

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平3-218524

⑬ Int. Cl.³

G 06 F 9/38

識別記号

3 3 0 D
3 3 0 F

庁内整理番号

7927-5B
7927-5B

⑭ 公開 平成3年(1991)9月26日

審査請求 未請求 請求項の数 4 (全13頁)

⑮ 発明の名称 命令処理装置

⑯ 特 願 平2-98102

⑰ 出 願 平2(1990)4月13日

優先権主張 ⑱ 平1(1989)11月8日 ⑲ 日本(JP) ⑳ 特願 平1-288832

㉑ 発 明 者 丸 島 敏 一 東京都港区芝5丁目7番1号 日本電気株式会社内

㉒ 出 願 人 日本電気株式会社 東京都港区芝5丁目7番1号

㉓ 代 理 人 弁理士 内 原 晋

明 細 書

発明の名称 命令処理装置

特許請求の範囲

(1) 分岐命令による分岐先の確定を待たずに分岐先アドレスを予測する分岐予測方式において、処理すべき命令を記憶する第1の手段と、命令の実行結果を書き込むレジスタという第2の手段と、前記第1の手段より得られた命令が実行結果として行うレジスタへの書き込みをあらかじめ予約する第3の手段と、前記第3の手段により予約された命令を保持し、命令を発行する第4の手段と、前記第4の手段により発行された命令を実行する第5の手段と、前記第5の手段と前記第2の手段との間に配置される、予測されていながら確定していない分岐先にある命令による実行結果を一時的に保持する第6の手段と、前記第6の手段により保持された実行結果を前記第2の手段へ転送する第7の手段と、前記第6の手段により保持された実行結果を前記第4の手段

がオペランドとして読み出す第8の手段と、前記第2の手段により保持された実行結果を前記第4の手段がオペランドとして読み出す第9の手段とを備え、さらに前記第4の手段は、分岐先未確定の分岐命令以前の命令系列を保持する第10の手段と、前記第10の手段に保持された命令と前記第3の手段により予約された命令を前記第5の手段に発行する第11の手段と、前記第3の手段によるレジスタ書き込み予約と前記第11の手段が行なう命令発行の履歴情報と前記第5の手段による分岐先確定情報とを受け取ることにより、前記第11の手段が行なうオペランドアクセスに対して、制御を加える第12の手段を備え、予測された確定前の分岐先の命令の実行結果を使用してさらにその先の命令の処理を行うことを特徴とする命令処理装置。

(2) 前記第10の手段を2つ以上備え、前記第3の手段により予約された命令を、分岐命令を区切りとして複数の第10の手段に対して、順次振り分けることを特徴とする特許請求の範囲第1項記載の命令処理装置。

(3) 前記第2の手段と第6の手段と第7の手段と第8の手段と第9の手段との代わりに、第2の手段と第9の手段の組を2つ以上と、その複数の第2の手段に保持されている値を互いに転送する手段とを備え、予測されているながら確定していない分岐先にある命令による実行結果を、複数の第2の手段のいずれかに保持することを特徴とする特許請求の範囲第1項記載の命令処理装置。

(4) 前記第11の手段を2つ以上備えることにより、複数の命令を同時に発行することを可能とすることを特徴とする特許請求の範囲第1項記載の命令処理装置。

発明の詳細な説明

(産業上の利用分野)

本発明は、分岐予測先の命令の実行結果を使用してさらにその先の命令の処理を行う命令処理装置に関するものである。

(従来の技術)

従来の命令処理装置では、分岐予測先の命令をフロッチ、デコードするもの(例えば、IBM360/91

命令処理装置: D. W. Anderson, F. J. Sparacio, F. M. Tomasulo: "The IBM System / 360 Model 91 : Machine Philosophy and Instruction Handling", IBM Journal of Research & Development, pp. 8-24, No. 1, Vol. 1, Jan. 1967.)や、分岐先が確定していてもオペランドが揃った命令は実行するがその実行結果は他の命令のソースオペランドにはならないもの(例えば、「新風」命令処理装置: 久我、村上、富田: "「新風」プロセッサの高速化メカニズム", 情報処理学会第37回全国大会講演論文集, 4N-2, 1988.)がある。これらの命令処理装置は、第16図に示すように、命令記憶手段10、レジスタ書込み予約手段20、命令発行手段40、オペランドアクセス制御手段50、命令実行手段60、分岐予測先命令実行結果格納手段70'及びレジスタ80を備えているが、後述の本願装置の分岐予測先命令実行結果格納手段からオペランド読出しをする手段を備えていない。

(発明が解決しようとする課題)

しかしながら、従来の命令処理装置では、予測した分岐先の命令実行は命令発行前の前処理のみであったり、予測先の命令を実行してもこの実行結果を使用する命令は分岐先が確定するまでは実行されなかった。このような制約を加えることは、本来実行可能な命令を待たせることになり、性能低下を及ぼすという問題点があった。

本発明の目的は、このような従来の問題点を除去して、分岐予測先の命令の実行結果を使用してさらにその先の命令の処理を行う命令処理装置を提供することにある。

(課題を解決するための手段)

前述の問題点を解決するために本願第1の発明は;

分岐命令による分岐先の確定を待たずに分岐先アドレスを分岐予測方式において、処理すべき命令を記憶する第1の手段と、命令の実行結果を書き込むレジスタという第2の手段と、第1の手段より得られた命令が実行結果として行なうレジスタへの書込みをあらかじめ予約する第3の手段と、第3

の手段により予約された命令を保持し、命令を発行する第4の手段と、第4の手段により発行された命令を実行する第5の手段と、第5の手段と第2の手段との間に配置される、予測されているながら確定していない分岐先にある命令による実行結果を一時的に保持する第6の手段と、第6の手段により保持された実行結果を第2の手段へ転送する第7の手段と、第6の手段により保持された実行結果を第4の手段がオペランドとして読み出す第8の手段と、第2の手段により保持された実行結果を第4の手段がオペランドとして読み出す第9の手段とを備え、さらに第4の手段は、分岐先未確定の分岐命令以前の命令系列保持する第10の手段と、第10の手段に保持された命令と第3の手段により予約された命令を第5の手段に発行する第11の手段と、第3の手段によるレジスタ書込み予約を受取り、第11の手段が行う命令発行の履歴情報を受取り、第5の手段による分岐先確定情報を受け取ることにより、第11の手段が行うオペランドアクセスに対して、制御を加える第12の手段を備え、予測された確定前の

分岐先の命令の実行結果を使用してさらにその先の命令の処理を行うことを特徴とする。

また本願第2の発明は;

前記第10の手段を2つ以上備え、第3の手段により予約された命令を、分岐命令を区切りとして複数の第10の手段に対して、順次振り分けることを特徴とする。

また本願第3の発明は;

前記第2の手段と第6の手段と第7の手段と第8の手段と第9の手段との代わりに、第2の手段と第9の手段の組を2つ以上と、その複数の第2の手段に保持されている値を互いに転送する手段とを備え、予測されているながら確定していない分岐先にある命令による実行結果を、複数の第2の手段のいずれかに保持することを特徴とする。

また本願第4の発明は;

前記第11の手段を2つ以上備えることにより、複数の命令を同時に発行することを可能とすることを特徴とする。

(作用)

定していない命令を発行する命令発行手段である。50は命令発行手段40が行うオペランドアクセスに対し制御を加えるオペランドアクセス制御手段である。60は命令発行手段40により発行された命令を実行する命令実行手段である。70は命令実行手段60から出力される結果の内、実行が確定していない命令による実行結果を一時的に保持する分岐予測先命令実行結果格納手段である。80は命令実行手段60から出力される結果の内、実行が確定している命令による実行結果を保持するレジスタである。

第2図はオペランドアクセス制御手段50の要部を示したものである。510はレジスタ書込み予約手段20により設定されるレジスタを書込み履歴フラグであり、命令分岐から次の分岐命令までを一つのブロックと考え、過去から現在を通してブロック内で該レジスタに対して書込み予約があったどうかを示す。520はレジスタ書込み予約手段20により設定されるレジスタ書込み予約済フラグであり、現時点でブロック内で該レジスタに対して書込み

このような手段をとることによって、分岐命令以前の命令系列と予測された分岐先の命令系列を可能な限り同時に実行しながら、予測された確定前の分岐先の命令による実行結果であってもオペランドとして使用することを可能とする。これにより、分岐命令の確定待ちによる命令実行開始の遅延を抑えることができ、高速な処理を可能とする。

(実施例)

第1図は本発明の一実施例を示す装置構成図である。図中10は処理すべき命令を記憶する命令記憶手段である。20は命令記憶手段10より得た命令が実行結果として行うレジスタへの書込みをあらかじめ予約するレジスタ書込み予約手段である。30はレジスタ書込み予約手段20により処理された命令を複数保持できる先入れ先出し機能を持つ命令保持手段であり、分岐先が確定し、実行されることが確定している命令系列を保持する。40は命令保持手段により保持された命令、もしくはレジスタ書込み予約手段20により処理された、実行が確

予約がされているかどうかを示す。530は命令発行手段40により設定される命令発行済フラグであり、現時点でブロック内で該レジスタに対して書込み予約をしている命令が発行されているかどうかを示す。レジスタ書込み履歴フラグ510、レジスタ書込み予約済フラグ520、命令発行済フラグ530は同一の形式で、第3図に示すように、レジスタの各要素に対応するフラグを持ち、さらに確定している命令用と確定していない命令用との2種類のフラグセットを備える。この2種類のフラグレジスタを区別するため、A側、B側という用語を用いる。命令分岐から次の命令分岐までの一つのブロックには、A側かB側のどちらか一方が割り当てられる。この割り当て該ブロックに含まれる全命令の終了時に解放され、次のブロックに割り当てることができるようになる。540はレジスタ書込み履歴フラグ510、レジスタ書込み予約済フラグ520、命令発行済フラグ530における2種類のフラグセットの内、どちら側が確定している命令用であることを示す確定命令系列フラグである。すなわち、現在

実行中のA側かB側のどちら側が実行が確定している命令系列なのかを示す。これにより、該ブロックが割り当てられている側が、確定命令系列540が規定する側と一致していれば該ブロックの実行は確定しているものと判定することができる。550命令発行手段40が行なうオペランドアクセスに対して情報を与えるオペランド読出し判定回路であり、オペランド読出しを分岐予測先命令実行結果格納手段70とレジスタ80のどちらから行うという読出し元の情報と、読出しを行って良いかという読出し可否の情報とを、レジスタ書込み履歴フラグ510、レジスタ書込み予約済フラグ520、命令発行済フラグ530、確定命令系列フラグ540による値から作成する。この読出し情報の判定流れ図を第4図に示す。第4図において、レジスタ書込み履歴フラグ510をWH、レジスタ書込み予約済フラグ520をWB、命令発行済フラグ530をIB、確定命令系列フラグ540をFP、該ブロック側をBlk、該ブロックの反対側をrBlk、判定する該レジスタ番号をRegで表す。特に、レジスタ書込み履歴フラグ510、レ

ジスタ書込み予約済フラグ520、命令発行済フラグ530については、例えば、「レジスタ書込み予約済フラグ520の、該ブロック側のフラグセットの、該レジスタ番号が示すフラグ」を、WB(Blk, Reg)という形で表す。

以上のような構成の命令処理装置において、本発明の内容を説明するために、まず実行されることが確定している命令を処理する際の動作を説明する。命令は命令記憶手段10から読み出され、その命令が実行結果として行うレジスタへの書込みを、レジスタ書込み予約手段20により予約する。レジスタ書込みの予約の予約操作では、レジスタ書込み履歴フラグ510とレジスタ書込み予約済フラグ520の両方に対して、該ブロック側フラグセットの該書込みレジスタ番号の示すフラグを設定する。ここで設定されたフラグの内、レジスタ書込み履歴フラグ510は該ブロックの終了時、すなわち次の分岐命令までの全命令が終了した時点で解除され、レジスタ書込み予約済フラグ520は該命令が終了した時点で解除される。レジスタ書込み予約

手段20により予約が完了した命令は、命令保持手段30に保持され、命令発行を待つ。命令保持手段30に保持されていた命令は、命令発行手段40においてオペランドアクセスを行ない、これに成功したものは命令実行手段60に転送される。ここで行われるオペランドアクセスにあたって、読出しに関する情報を、オペランドアクセス制御手段50より受け取る。このときオペランドアクセス制御手段50では、第4図の流れ図に沿って判定が下される。まず、該ブロックと確定命令系列フラグ540の比較が行われ(ステップ101)、これが一致していることから、該ブロックは実行されることが確定していることがわかる。そして、命令発行済フラグ530の該ブロック側フラグセットの該読出しレジスタ番号の示すフラグを判定し(ステップ102)、このフラグが解除された状態であれば、レジスタ80への該オペランドアクセスを許可する。命令発行手段40から命令を発行すると同時に、命令発行済フラグ530の該ブロック側フラグセットの該書込みレジスタ番号の示すフラグを設定する。命令発行手

段40から発行された命令は命令実行手段60において該命令が規定する動作を行ない、その実行結果を生成する。命令実行手段60において生成された実行結果は、レジスタ80に格納され、実行を終了する。命令の実行終了時には、レジスタ書込み予約済フラグ520と命令発行済フラグ530の両方に対して、該ブロック側フラグセットの該書込みレジスタ番号の示すフラグを解除する。

次に、分岐予測機構により予測された、実行が確定していない命令を実行する場合の動作について説明する。命令は命令記憶手段10から読み出され、その命令が実行結果として行うレジスタへの書込みを、レジスタ書込み予約手段20により予約する。レジスタ書込みの予約操作では、レジスタ書込み履歴フラグ510とレジスタ書込み予約済フラグ520の両方に対して、該ブロック側の該書込みレジスタ番号の示すフラグを設定する。ここで設定されたフラグの内、レジスタ書込み履歴フラグ510は次の分岐命令までの全命令が終了した時点で解除され、レジスタ書込み予約済フラグ520は該命令

が終了した時点で解除される。レジスタ書込み予約手段20により予約が完了した命令は、命令発行手段40においてオペランドアクセスを行ない、これに成功したものは命令実行手段60に転送される。ここで行われるオペランドアクセスにあたって、オペランド読出しを分岐予測先命令実行結果格納手段70とレジスタ80のどちらから行うかという読出し元の情報と、読出しを行って良いかという読出し可否の情報とを、オペランドアクセス制御手段50より受け取る。このときオペランドアクセス制御手段50では、第4図の流れ図に沿って判定が下される。まず、該ブロックと確定命令系列フラグ540の比較が行われ(ステップ101)、これが一致していないことから、該ブロックは実行されることが確定しないことがわかる。次にレジスタ書込み履歴フラグ510の該ブロック側フラグセットの該読出しレジスタ番号の示すフラグを判定し(ステップ103)、このフラグが解除された状態であればオペランドアクセスをレジスタ80から、設定された状態であればオペランドアクセスを分岐予測先命令

実行結果格納手段70から行うように、命令発行手段40に伝える。さらにこのオペランド読出し元情報を元に、オペランドアクセス制御手段50が命令発行手段40に対して伝える、オペランドアクセス許可情報を作成する。すなわち、もしレジスタ80から読み出す場合であれば、レジスタ書込み予約済フラグ520の該ブロックの反対側のフラグセットの該読出しレジスタ番号の示すフラグを判定し(ステップ104)、このフラグが解除された状態であればレジスタ80への該オペランドアクセスを許可する。もし、分岐予測先命令実行結果格納手段70から読み出す場合であれば、命令発行済フラグ530の該ブロック側フラグセットの該レジスタ番号の示すフラグを判定し(ステップ105)、このフラグが解除された状態であれば分岐予測先命令実行結果格納手段70への該オペランドアクセスを許可する。命令発行手段40から命令を発行すると同時に、命令発行済フラグ530の該ブロック側フラグセットの該書込みレジスタ番号の示すフラグを設定する。命令発行手段40から発行された命令は命令実行手

段60において該命令が規定する動作を行ない、その実行結果を生成する。命令実行手段60において生成された実行結果は、分岐予測先命令実行結果格納手段70に一時的に保持される。この時点でとりあえず命令は終了したものとみなされ、レジスタ書込み予約済フラグ520と命令発行済フラグ530の両方に対して、該ブロック側の該書込みレジスタ番号の示すフラグを解除する。

次に、分岐命令による分岐先が確定し、予測した分岐先と一致していた時の動作を説明する。分岐命令の分岐先が確定したら、その分岐命令以前の命令の実行終了を待った後、分岐予測先命令実行結果格納手段70に一時的に保持されていた実行結果をレジスタ80に転送する。すべての転送が終了後、分岐予測先命令実行結果格納手段70の全要素を解除する。レジスタ書込み履歴フラグ510では、確定命令系列フラグ540が規定するフラグセットを全要素解除する。その後、確定命令系列フラグ540を反転させる。

次に、分岐命令による分岐先が確定したが、予測した分岐先と一致せず、分岐予測が失敗した時の動作を説明する。分岐予測失敗を検知すると、分岐予測失敗した先の命令の命令記憶手段10からの投入を停止し、命令実行手段40にすでに投入されている分岐予測失敗した先の命令の実行の終了を待ち、分岐予測先命令実行結果格納手段70の全要素を解除する。レジスタ書込み履歴フラグ510、レジスタ書込み予約済フラグ520、命令発行済フラグ530の全てにおいて、確定命令系列フラグ540が規定する側の逆側のフラグセットを全要素解除する命令記憶10からは確定した分岐先の命令を投入し、実行が確定した命令として実行を続行する。

次に第5図に示すプログラムにおいて動作例を説明する。プログラム中の分岐命令については分岐を起こさない方に予測されているものとする。また、確定命令系列フラグ540は初期値としてA側を指示するものとする。つまり、最初に投入される命令から次の分岐命令までは、レジスタ書込み履歴フラグ510、レジスタ書込み予約済フラグ520、

命令発行済フラグ530におけるA、B2種類のフラグセットの内、A側を使用する。まず第1サイクルでは命令(1)が命令記憶10から読み出されて、レジスタ書き込み予約手段20を通じて、命令(1)の書き込みレジスタr2に対応した、レジスタ書き込み履歴フラグ510とレジスタ書き込み予約済フラグ520の両者のA側の2番のフラグをONにする。第2サイクルでは命令(2)が命令記憶手段10から読み出されて、レジスタ書き込み予約手段20を通じて、命令(2)の書き込みレジスタr4に対応した、レジスタ書き込み履歴フラグ510とレジスタ書き込み予約済フラグ520の両者のA側の4番のフラグをONにする。それと同時に、命令(1)が命令保持手段30により保持される。第3サイクルでは、命令(3)が命令記憶手段10から読み出されるが、命令(3)はレジスタ書き込みがないので、レジスタ書き込み予約手段20は書き込み予約はしない。それと同時に命令(2)が命令保持手段30により保持される。それと同時に命令発行手段40において命令(1)がオペランドr0とr1を読み出し、命令を発行する。この時、命令発行済フラグのA側の0番と1番フ

ラグをチェックし、共にOFFであるのでオペランドアクセスは成功する。命令発行時には、命令(1)の書き込みレジスタr2に対応した、命令発行済フラグ530の両者のA側の2番のフラグをONにする。第4サイクルでは、命令(4)が命令記憶手段10から読み出されて、レジスタ書き込み予約手段20を通じて、命令(4)の書き込みレジスタr5に対応した、レジスタ書き込み履歴フラグ510とレジスタ書き込み予約済フラグ520の両者のB側の5番のフラグをONにする。それと同時に命令(3)が命令保持手段30により保持される。それと同時に命令発行手段40において命令(2)がオペランドr2とr3を読み出そうとするが、命令発行済フラグのA側の2番のフラグがONになっているため、オペランドアクセスは失敗し、命令発行手段40に留まる。第4サイクルにおけるスナップショットを第6図に示す。

第5サイクルでは、命令(1)は演算実行を続け、命令(2)は命令発行手段40に、命令(3)は命令保持手段30に保持され続ける。命令(5)が命令記憶手段10から読み出されて、レジスタ書き込み予約手段20を通

じて、命令(5)の書き込みレジスタr6に対応した、レジスタ書き込み履歴フラグ510とレジスタ書き込み予約済フラグ520の両者のB側の6番のフラグをONにする。命令(4)は、まず命令(4)の読出しレジスタr0、r3に対応した、レジスタ書き込み履歴フラグ510のB側の0番と3番を判定し、共にOFFであるので共にレジスタ80からオペランドを読み出すことになる。そして、レジスタ書き込み予約済フラグ520のA側の0番と3番を判定し、共にOFFであるのでオペランドアクセスは成功し、命令発行手段40から命令(4)が発行される。第6のサイクルでは命令(6)が命令記憶手段10から投入される。それと同時に命令発行手段40において命令(5)がオペランドr2とr3を読み出そうとするが、命令発行済フラグのA側の2番のフラグがONになっているため、オペランドアクセスは失敗し、命令発行手段40に留まる。第7サイクルでは命令(7)が命令記憶手段10から投入されようとするが、レジスタ書き込み予約手段20に命令(6)が留まっているため、投入できない。第8サイク

ルでは変化がない。第8サイクルにおけるスナップショットを第7図に示す。

第9サイクルでは命令(1)の実行結果のレジスタ80への書き込みが起こり、レジスタ書き込み予約済フラグ520と命令発行済フラグのA側の2番をOFFにする。これにより、命令発行済フラグのA側の2番がOFFになるのを待っていた、命令発行手段40にある命令(2)と命令(5)の両者のオペランドアクセスが成功するが、本実施例では同時には1つの命令しか発行できない構成になっているため、確定系列側の命令(2)のみが発行される。第10サイクルでは命令(3)の発行が試みられるが、命令(2)によって命令発行済フラグのA側の4番がONにされているためオペランドアクセスに失敗し、命令発行手段40に留まる。また第10サイクルでは前サイクルに保留された命令(5)の発行が行われる。これによって、第11サイクルでは命令(6)が命令発行手段40に、命令(7)がレジスタ書き込み予約手段20に進む。また、命令(4)の実行結果の分岐予測先命令実行結果格納手段70への書き込みが起こり、レジスタ書き込み予約済

フラグ520と命令発行済フラグ520のB側の5番をOFFにする。これにより、以下のように命令発行手段40にある命令(6)のオペランドアクセスが成功し、命令(6)が発行される。すなわち、レジスタ書込み履歴フラグ510のB側の5番がONになっているため、オペランド読出し分岐予測先命令実行結果格納手段70からであることがわかり、命令発行済フラグ530のB側の5番がOFFになったことによりオペランドアクセスが許可されたことになる。第11サイクルにおけるスナップショットを第8図に示す。

第12サイクルから第14サイクルの間は変化が起らない。第15サイクルでは命令(2)の実行結果のレジスタ80への書込みが起り、レジスタ書込み予約済フラグ520と命令発行済フラグのA側の4番をOFFにする。これにより、命令発行済フラグのA側の4番がOFFになるのを待っていた、命令発行手段40にある命令(3)が発行される。第16サイクルでは命令(2)の、そして第17サイクルでは命令(6)の実行結果の分岐予測先命令実行結果格納手段70への書

込みが起り、レジスタ書込み予約済フラグ520と命令発行済フラグ530のB側の6番と7番をOFFにする。これにより、第17サイクルでは、命令発行済フラグのB側の7番がOFFになるのを待っていた、命令発行手段40にある命令(7)が発行される。第18サイクルでは変化がない。第19サイクルでは、命令(3)の実行が終了し、予測した分岐先に確定するので、分岐予測先命令実行結果格納手段70からレジスタ80への転送を行う。このとき分岐予測先命令実行結果格納手段70内には、命令(4),(5),(6)による実行結果r5,r6,r7があり、転送はこの3つについて行われる。転送後、分岐予測先命令実行結果格納手段70は解除される。そして、レジスタ書込み履歴フラグのA側を全要素解除し、確定命令系列フラグ540の規定する値をA側からB側に反転する。これにより、次の第20サイクルではA側が新たに使用できるようになるので命令(8)が命令記憶10から投入される。第20サイクルにおけるスナップショットを第9図に示す。

第21サイクルでは、命令(7)の実行が終了し、予測した分岐先に確定する。但し、このとき分岐予測先命令実行結果格納手段70内には書込みが起きていないので、分岐予測先命令実行結果格納手段70からレジスタ80への転送は行われない。そして、レジスタ書込み履歴フラグのB側を全要素解除し、確定命令系列フラグ540の規定する値をB側からA側に反転する。命令(8)は第21サイクルで発行され、第26サイクルで終了し、プログラムが終了する。プログラムの開始から終了までのタイムチャートを第10図に示す。

以上、本願発明による命令処理装置において、命令の処理過程について説明した。

次に、第1図の1つの命令保持手段30の代わりに2つの命令保持手段31,32を生じた場合の本発明の他の一実施例を第11図により説明する。

第11図は、本発明による命令処理装置の一実施例である。図において2つの命令保持手段31,32を備えている以外は、前述した第1図の本発明による命令処理装置の実施例と同じ構成となっている。

本構成では、レジスタ書込み予約手段20により予約が完了した命令は、分岐命令を区切りとして、命令保持手段AもしくはBに保持される。例えば、初めに命令保持手段Aに保持されていた命令系列では、分岐命令の次の命令以降を命令保持手段Bに保持し、またその次の分岐命令の次の命令以降を命令保持手段Aに保持していく。これにより、第5図のプログラムの例では、命令(7)は第10サイクルまで命令記憶10から投入できなかったが、命令保持手段を2つ使用した本発明では第6サイクルにおいて投入が可能となる。レジスタを2つ使用した場合の本発明のさらに他の一実施例を第12図により説明する。

第12図の本発明の命令処理装置は2つのレジスタ81,82とその間の転送手段90を備えており、分岐予測先命令実行結果格納手段70が省かれている以外は、前述した第1図の本発明による命令処理装置の実施例と同じ構成となっている。本構成では、命令実行手段60にて得られた結果は、確定している系列に属する命令による結果であればレジスタA,B

の両者に書込むが、確定していない系列による結果であれば、分岐命令を区切りとして、レジスタAもしくはBのいずれかに保持される。例えば、ある確定していない命令系列による結果をレジスタAに保持していった場合には、先行する分岐予測が成功してこの命令系列が確定した瞬間からこの系列による結果をレジスタA,Bの両者に書込む。そして、分岐命令の次の命令以降の結果は実行が確定するまでレジスタBだけに保持し、その命令系列が確定した瞬間からその系列による結果をレジスタA,Bの両者に書込む。またその次の分岐命令の次の命令以降の結果を実行が確定するまではレジスタAに保持していき、以下これを繰り返す。レジスタ間転送手段90では、レジスタ書込み済フラグという前述の第3図の形式のフラグレジスタを備え、上記の動作手順中で片方のレジスタにのみ結果が書き込まれた場合には、該当する側(すなわち、レジスタAならA側、レジスタBならB側)の該レジスタ番号のレジスタ書込み済フラグをONにする。そして、その命令系列の実行可否を決定する分岐命令

が予測成功側に分岐した時点で、レジスタ書込み済フラグがONとなっている該ブロック側の該レジスタ番号のレジスタを、反対側の同一レジスタ番号に複写する。レジスタ書込み済フラグは、この複写後OFFとする。また、逆に予測失敗側に分岐した場合には、レジスタ書込み済フラグがONとなっている該ブロック側の反対側の該レジスタ番号のレジスタを、レジスタ書込み済フラグがONとなっている該ブロック側の同一レジスタ番号に複写する。レジスタ書込み済フラグは、この複写後OFFとする。これにより、分岐予測先の実行結果を分岐先が確定する前に片側のレジスタに書込み、分岐成功時にはその結果を反対側に複写し、分岐予測失敗時にはその反対側の値を複写してることにより元の値を復元することができる。

第12図の実施例における、レジスタ読出しの可否と、いずれのレジスタから読出すべきかの判断を決定する流れ図を第13図に示す。使用する記号は第4図と同様である。

次に第12図の実施例における動作を、前述の第5図に示すプログラムにおいて説明する。前述の第1図の場合に比べて、オペランド読出しと実行結果書込みの動作だけが異なるため、この部分についてのみ説明する。まず、命令(1)は確定系列側であり(第13図、ステップ201)、r2の書込みをする命令が実行中ではない(ステップ204)ので、第13図の流れ図に従いA側のr2を第3サイクルで読出すことが可能である。また、命令(1)は書込み時には確定系列であるので両側のレジスタに書き込む。次の命令(2)はやはり確定系列側であるので、第13図の流れ図に従い該ブロック側、すなわちA側のレジスタからオペランド読出しをする。この際、先行する命令(1)が、既にr2の書込みを予約している(すなわち、IB(Blk, Reg)がON)ため、命令(1)が終了する第9サイクルまでオペランド読出しが保留される。命令(5)は命令(2)と同じくr2を読み出そうとするが、確定していない系列側であり(第13図、ステップ201)、該ブロックによるr2への書込みがされていない(ステップ202)ため、該ブロックの反対側、すな

わちA側のレジスタからオペランド読出しを試みる。ところが命令(1)がすでにr2の書込みを予約している(ステップ203)ため、命令(1)が終了するまでオペランド読出しが保留される(第9サイクルでは命令(2)が発行されるため、命令(5)の発行は第10サイクル)。また、命令(5)は書込み時にも確定していない系列側であるため、該ブロック側、すなわちB側のレジスタにのみ結果の書込みをする。その他の命令についても同様の動作をする。

次に、命令発行手段を2つ使用した場合の一実施例を第14図、第15図により説明する。第14図は、本発明による命令処理装置の一実施例である。図において、2つの命令発行手段41,42備え、複数命令を同時発行可能としており、これに対処するためにオペランドアクセス制御手段150、命令実行手段160、分岐予測先命令実行結果格納手段170、レジスタ180が同時アクセス可能な構成になっている以外は、前述した第1図の本発明による命令処理装置の実施例と同じ構成となっている。また第15図はオペランドアクセス制御手段150の要部を示した

ものである。図において、各フラグが同時アクセス可能になり、オペランド読出し判定回路を2つ備えている以外は、前述した第2図と同じ構成となっている。

次に第14図の実施例における動作を、前述の第5図に示すプログラムにおいて説明する。前述の第1図の場合に比べ、命令発行手段の動作タイミングだけが異なるため、この部分についてのみ説明する。第9サイクルでは命令(1)の実行結果のレジスタ80への書込みが起こり、レジスタ書込み予約済フラグ520と命令発行済フラグのA側の2番をOFFにする。これにより、命令発行済フラグのA側の2番OFFになるのを待っていた、命令発行手段にある命令(2)と命令(5)の両者のオペランドアクセスが成功する。第1図の実施例では同時には1つの命令しか発行できない構成になっているため、確定系列側の命令(2)のみが発行されたが、第14図による実施例では命令発行手段を2つ備えて2命令同時発行可能であるため、第9サイクルに命令(2)と命令(5)の

両者を発行する。他の命令の動作については第1図の実施例の場合と同じである。

なお、第11図、第12図、第14図の各実施例については、これらを適宜組み合わせて命令処理装置を構成してもよい。すなわち、複数の命令保持手段A、B、レジスタA、B及び命令発行手段A、Bのうちの少なくとも2つの手段を含むように命令処理装置を構成してもよい。

(発明の効果)

以上の説明で明かなように、この発明によると分岐命令以前の命令系列と予測された分岐先の命令系列を可能な限り同時に実行しながら、予測された確定前の分岐先の命令による実行結果であってもオペランドとして使用することが可能となり、分岐命令の確定待ちによる命令実行開始の遅延を抑えることができ、高速な処理を行うことができる。

図面の簡単な説明

第1図は本発明の一実施例を示す装置図、第2図は第1図のオペランドアクセス制御手段の一実施例

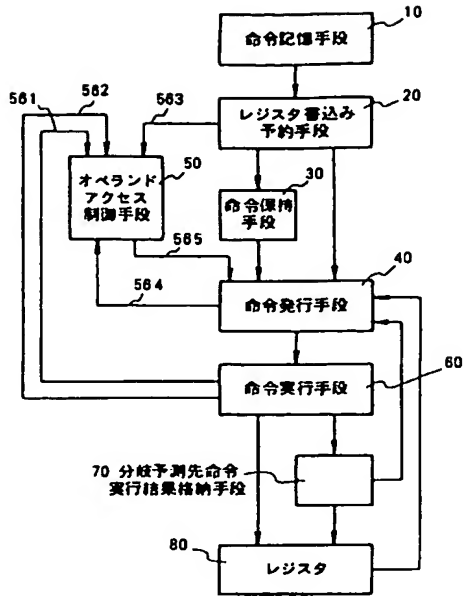
を示す図、第3図は第2図のレジスタ書込み履歴フラグ、レジスタ書込み予約済フラグ、命令発行済フラグ、第12図のレジスタ間転送手段のレジスタ書込み済フラグの構成を示す図である。第4図は第1図のオペランドアクセス制御手段における読出し許可判定のための流れ図である。第5図は動作例を示すために用いたプログラムである。第6, 7, 8, 9図はそれぞれ第5図によるプログラム動作例の第4, 8, 11, 20サイクルのスナップショットである。第10図は第5図によるプログラム動作例のタイムチャートである。第11図は本願の第2の発明の一実施例を示す装置図である。第12図は本願の第3の発明の一実施例を示す装置図である。第13図は第12図のオペランドアクセス制御手段における読出し許可判定のための流れ図である。第14図は本願の第4の発明の一実施例を示す装置図であり、第15図は第14図のオペランドアクセス制御手段の一実施例を示す図であり、第16図は従来技術による装置図である。

図において、

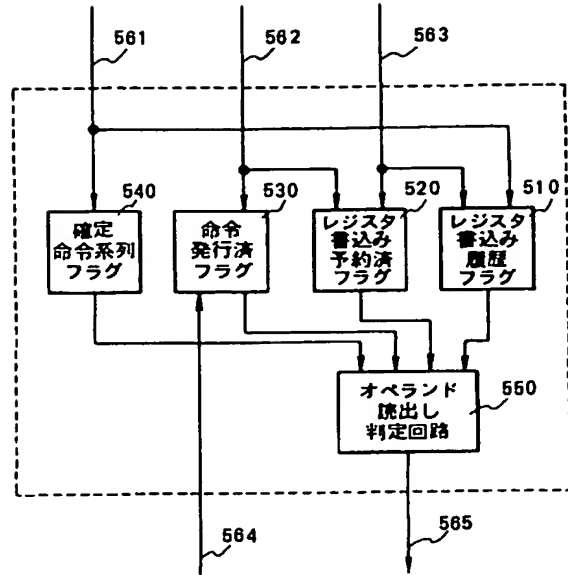
10は命令記憶手段、20はレジスタ書込み予約手段、30, 31, 32は命令保持手段、40, 41, 42は命令発行手段、50はオペランドアクセス制御手段、60は命令実行手段、70は分岐予測先命令実行結果格納手段、80, 81, 82はレジスタ、90はレジスタ間転送手段、150はオペランドアクセス制御手段、160は命令実行手段、170は分岐予測先命令実行結果格納手段、180はレジスタ、510はレジスタ書込み履歴フラグ、520はレジスタ書込み予約済フラグ、530は命令発行済フラグ、540は確定命令系列フラグ、550はオペランド読出し判定回路、561は分岐命令の分岐情報を伝える信号線、562は各命令の終了を伝える信号線、563は命令のレジスタへの書込み予約情報を伝える信号線、564は各命令の発行情報を伝える信号線、565はオペランド読出しの可否を伝える信号線、1510はレジスタ書込み履歴フラグ、1520はレジスタ書込み予約済フラグ、1530は命令発行済フラグ、1540は確定命令系列フラグである。

代理人 弁理士 内原 晋

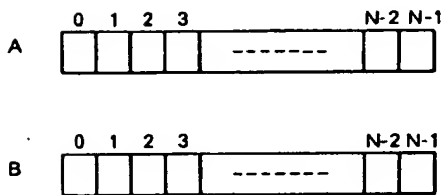
第 1 図



第 2 図

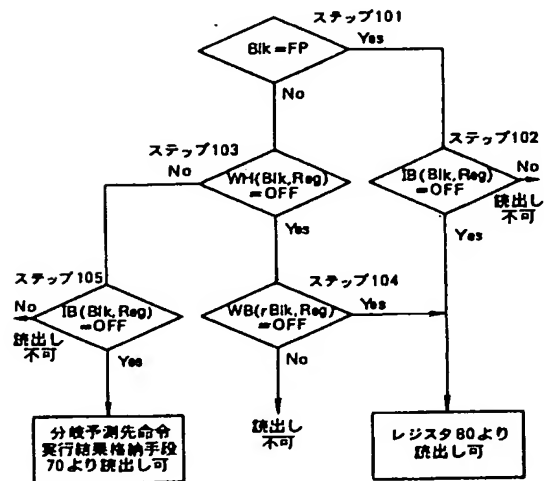


第 3 図



N : レジスタの要素数

第 4 図



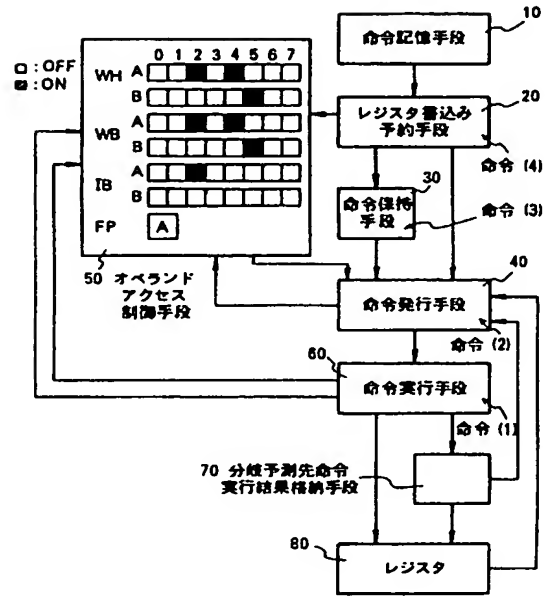
WH : レジスタ書込み履歴フラグ510
WB : レジスタ書込み予約済フラグ520
IB : 命令発行済フラグ530
FP : 確定命令系列フラグ540
Blk : 該ブロック側
rBlk : 該ブロックの反対側
Reg : 該レジスタ番号

第 5 図

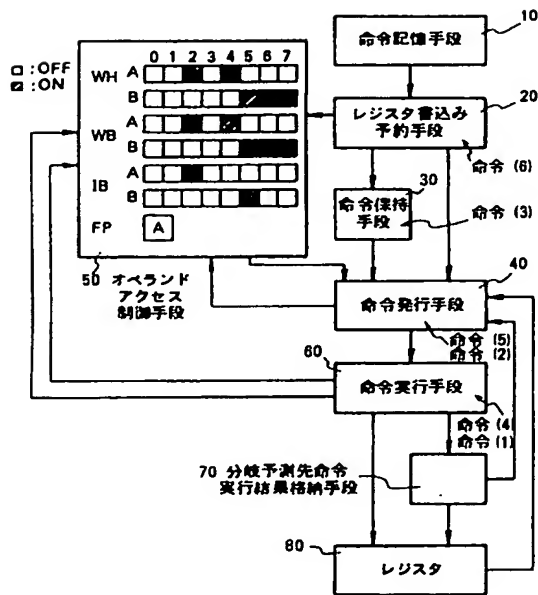
命令発行後実行時間

命令 (1) : 乗算命令	Mlt r0*r1→r2	8 サイクル
命令 (2) : 乗算命令	Mlt r2*r3→r4	8 サイクル
命令 (3) : 分岐命令	BrchTo X1 on (r4 = 0)	4 サイクル
命令 (4) : 乗算命令	Mlt r0*r3→r5	6 サイクル
命令 (5) : 乗算命令	Mlt r2*r3→r6	6 サイクル
命令 (6) : 乗算命令	Mlt r0*r5→r7	6 サイクル
命令 (7) : 分岐命令	BrchTo X2 on (r7 = 0)	4 サイクル
命令 (8) : 加算命令	Add r0+r3→r0	5 サイクル

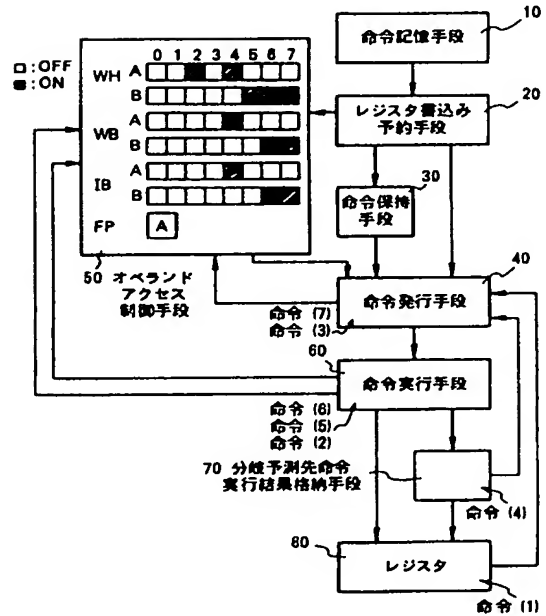
第 6 図



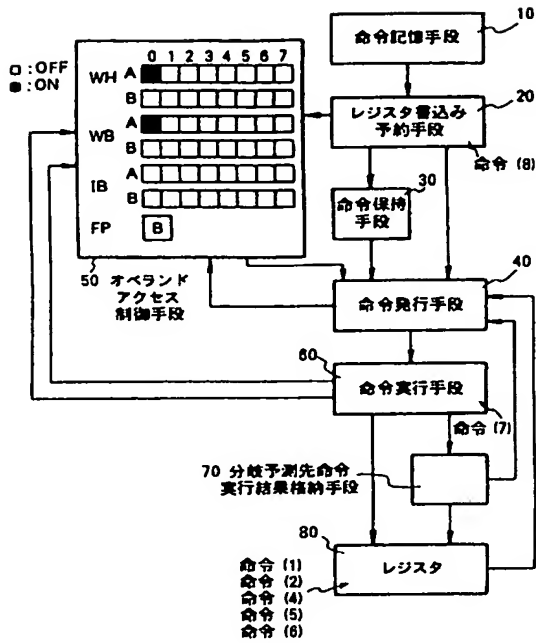
第 7 図



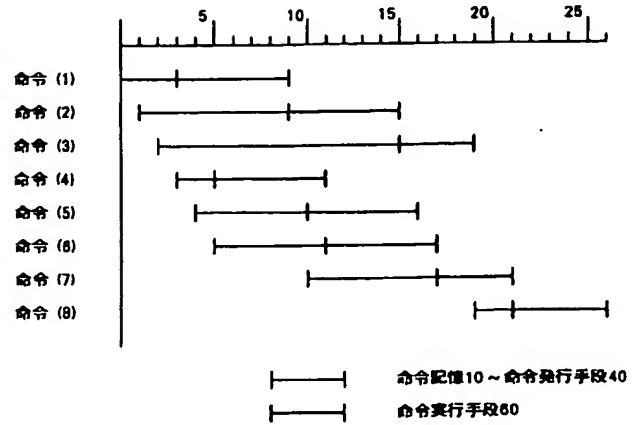
第 8 図



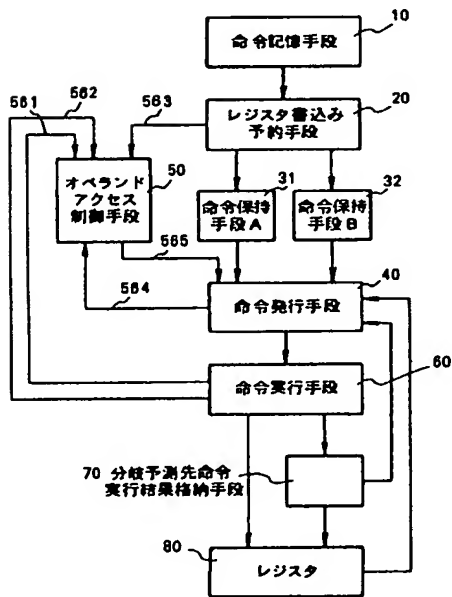
第 9 図



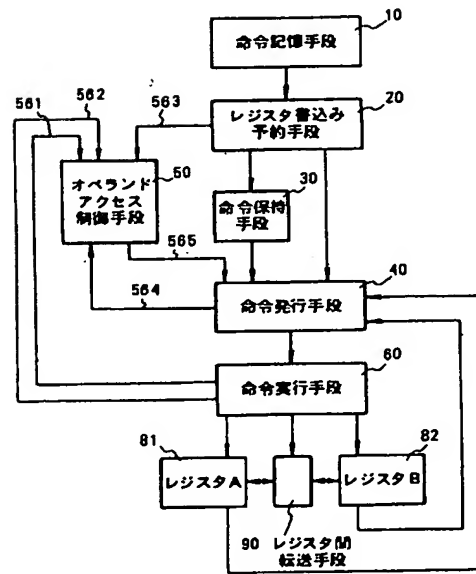
第 10 図



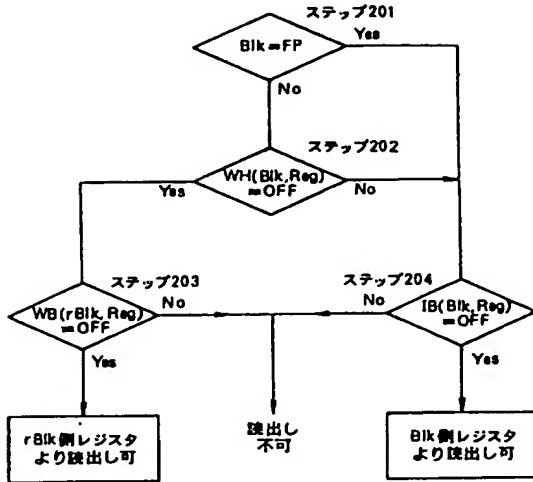
第 11 図



第 12 図

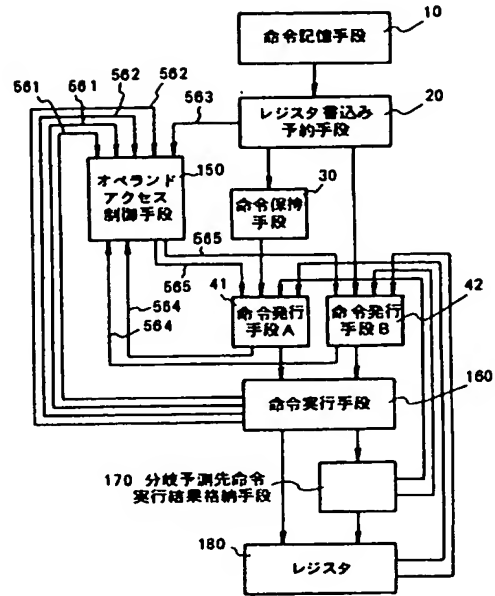


第 13 図

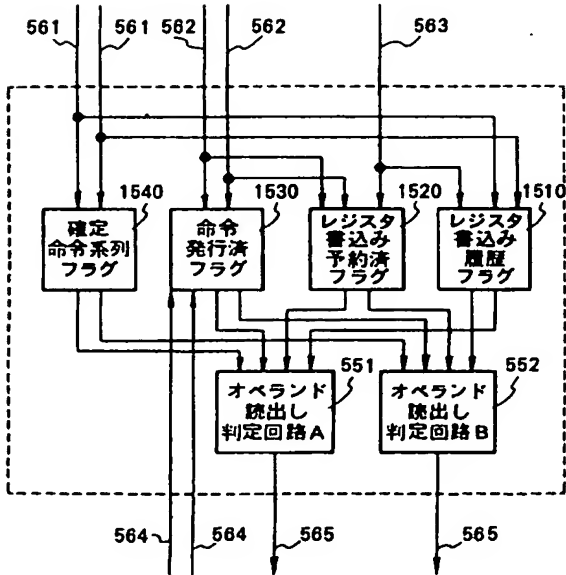


WH : レジスタ書込み履歴フラグ610
WB : レジスタ書込み予約済フラグ620
IB : 命令発行済フラグ630
FP : 確定命令系列フラグ640
Blk : 該ブロック側
rBlk : 該ブロックの反対側
Reg : 該レジスタ番号

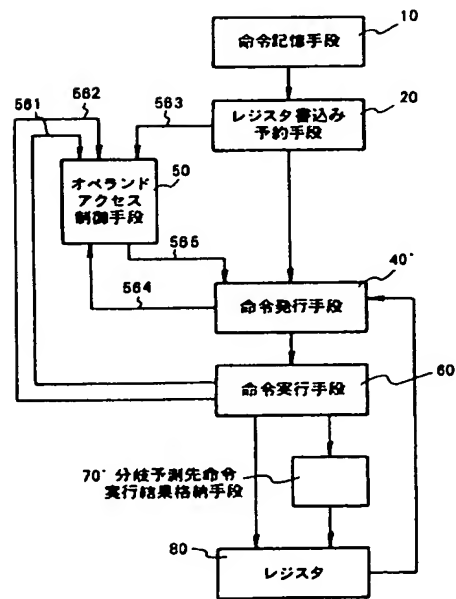
第 14 図



第 15 図



第 16 図



JAPANESE LAID-OPEN PATENT APPLICATION

H3-218524 (1991)

(19) Japan Patent Office (JP)

(11) Publication No. H3-218524

(12) Laid-Open Patent Application (A)

(43) Publication Date September 26, 1991

(51) Int. Cl.⁵

Identification Code

In-House Reference. No.

G 06F 9/38

330

D

7927-5B

330

F

7927-5B

No examination request

Number of claims 4

(Altogether 13 pages)

(54) Title of the Invention

INSTRUCTION PROCESSOR

(21) Application No.

PA H2-98102

(22) Date of Filing

April 13, 1990 (Heisei 2)

(32) Priority Claim Date

November 8, 1989 (Heisei 1)

(33) Priority Claim Country

Japan (JP)

(31) Priority Claim No.

PA H1-288832

(72) Inventor

Toshiichi MARUSHIMA

NEC CORP.

5-7-1, Shiba, Minato-ku

Tokyo

(71) Applicant

NEC Corp.

5-7-1, Shiba, Minato-ku

Tokyo

(74) Agent

Susumi UCHIHARA, Attorney

Specification

I. Title of the Invention

Instruction Processor

II. Claims

Claim (1) An instruction processor, characterized by the fact that a branching prediction mode for predicting a branching destination address without waiting for the confirmation of the branching destination with a branching instruction is provided with:

a first means for storing an instruction to be processed,
a second means called register for writing the execution result of the instructions,
a third means for preliminarily reserving a writing for executing the instruction obtained from the first means into the register as the execution result,
a fourth means for holding the instructions reserved by the third means and issuing an instruction,
a fifth means for executing the instruction issued by the fourth means,
a sixth means disposed between the fifth means and second means and tentatively holding the execution result based on an instruction in an unconfirmed but predicted branching destination,
a seven means for transferring the execution result held by the sixth means to the second means,
an eighth means for reading the execution result held by the sixth means as an operand taken by the fourth means, and
a ninth means for reading the execution result held by the second

means as an operand taken by the fourth means, wherein the fourth means is further provided with a tenth means for holding an instruction group before the branching instruction whose branching destination is unconfirmed, an eleventh means for issuing the instruction held in the tenth means and the instruction reserved by the third means, and a twelfth means for applying a control to the operand access obtained by the eleventh means by receiving the historical information of the register writing reservation given by the third means and the execution issuance made by the eleventh means and the confirmation information of branching destination given by the fifth means, wherein the execution results of the instructions of predicted branching destination before confirmation is used to further perform the processing of the instructions relating to that destination.

Claim (2)

The instruction processor according to Claim 1, characterized by providing two or more of the tenth means and distributing the instructions reserved by the third means to the plural tenth means with the branching instructions as marks.

Claim (3)

The instruction processor according to Claim 1, characterized by providing two or more sets of the second means and ninth means and a means for transferring values held in the plural second means in place of the second means, sixth means, seventh means, eighth means and ninth means and holding the execution results based on the

instruction in an unconfirmed but predicted branching destination in either of the plural second means.

Claim (4)

The instruction processor according to Claim 1, characterized by enabling the issuance of plural instructions simultaneously by providing two or more of the eleventh means.

[Detailed description of the invention]

[Field of industrial application]

The present invention relates to an instruction processor in which the execution results of an instruction of a predictive branching destination is used to further perform the processing of the instructions relating to the destination.

In conventional instruction processors, a processor in which an instruction of branching predictive destination is fetched and decoded (e. g., IBM360/91 instruction processor: D. W. Anderson, F. J. Sparacio and F. M. Tomasulo: "The IBM System/360 Model 91: Machine Philosophy and Instruction Handling", *IBM Journal of Research & Development*, Vol. 1, No. 1, 8 -24 (Jan., 1967) and a processor in which an instruction with uniform operands is executed even if a branching destination is unconfirmed, but its execution result cannot be the source operands of another instruction (e. g., [Shinfu] instruction processor: Kuga, Murakami and Fumida: "Acceleration mechanism of [Shinfu] instruction processor", *Symposium of the 37th National Meeting of the Society*

of Information Processing, 4N2, (1988) are given. As shown in Fig. 16, the instruction processors are provided with an instruction storage means 10, a register writing reserving means 20, an instruction issuing means 40', an operand access controlling means 50, an instruction executing means 60, and a branching predictive destination instruction execution result storage means 70' and a register 80 but are not provided with the means of the processor of the application described later for making the operand reading from the branching predictive destination instruction execution result storage means.

[Problems overcome by the invention]

In conventional instruction processors, however, the instruction execution of a predicted branching destination has only comprised preprocessing before the issuance of an instruction, and the execution results have not been executed until the confirmation of a branching destination for an instruction using the execution results even if an instruction of predictive destination has been executed. The application of such a limitation had the problem that an originally executable instruction waited to cause property deterioration.

The purpose of the present invention consists of eliminating such prior problems and providing an instruction processor in which the execution results of an instruction of a branching predictive destination is used to further perform the processing of the destination instructions.

[Problem resolution means]

In order to solve these problems, the first invention of the present application is characterized by the fact that a branching prediction mode for predicting a branching destination address without waiting for the confirmation of the branching destination by means of a branching instruction is provided with a first means for storing an instruction to be processed,
a second means called register for writing an execution result of the instructions,
a third means for preliminarily reserving a writing for executing the instruction obtained from the first means into the register as execution result,
a fourth means for holding the instructions reserved by the third means and issuing an instruction,
a fifth means for executing the instruction issued by the fourth means,
a sixth means disposed between the fifth means and second means and tentatively holding the execution result based on an instruction in an unconfirmed but predicted branching destination,
a seven means for transferring the execution results held by the sixth means to the second means,
an eighth means for reading the execution results held by the sixth means as an operand taken by the fourth means, and
a ninth means for reading the execution result held by the second means as an operand taken by the fourth means,
the fourth means being further provided with a tenth means for holding an instruction group before receiving the branching instruction whose branching destination is unconfirmed, an eleventh means for issuing the instruction held in the tenth means and the instruction reserved by the third means, and

a twelfth means for applying a control to the operand access obtained by the eleventh means by receiving history information of the register writing reservation given by the third means and the execution issuance made by the eleventh means and confirmation information of the branching destination given by the fifth means, and the execution results of the instructions of the predicted branching destination before confirmation is used to further perform the processing of the instructions of that destination.

The second invention of the present application is characterized by providing two or more of the tenth means and distributing the instructions reserved by the third means to the plural tenth means with the branching instructions as marks.

The third invention of the present application is characterized by providing two or more sets of the second means and ninth means and a means for transferring values held in the plural second means in place of the second means, sixth means, seventh means, eighth means and ninth means and holding the execution results based on the instruction in an unconfirmed but predicted branching destination in either of the plural second means.

The fourth invention of the present application is characterized by enabling the issuance of plural instructions simultaneously by providing two or more of the eleventh means.

[Functions]

The present invention enables the use of an instruction group before branching instructions and an instruction group of predicted branching destinations as operands while executing them simultaneously as far as possible even if they are execution results based on instructions of predicted branching destinations before confirmation. It can suppress the delay of instruction execution start caused by waiting for the confirmation of the branching destinations and enables high speed processing.

[Embodiments]

Fig. 1 is a system block diagram showing one Embodiment of the present invention. In the diagram, 10 is an instruction storage means for storing instructions to be processed. 20 is a register writing reserving means for preliminarily reserving a writing made by the instruction obtained from the instruction storage means 10 into the register as execution result. 30 is an instruction holding means which can hold multiple instructions processed by the register writing reserving means 20 and holds the first-in/first-out function and holds an instruction group which confirms that branching destinations are confirmed and executed. 40 is an instruction issuing means for issuing an instruction held by the instruction holding means 30 or an instruction which is processed by the register writing reserving means 20 and whose execution has not been confirmed. 50 is an operand access controlling means for applying a control to an operand access got by the instruction issuing means 40. 60 is an instruction executing means for executing an instruction issued by the instruction issuing means 40. 70 is a branching predictive destination instruction execution result storage means

for tentatively holding an execution result based on an instruction whose execution has not been confirmed. 80 is a register for holding an execution result based on an instruction whose execution has been confirmed in a result output from the instruction executing means 60.

Fig. 2 shows principal parts of the operand access controlling means 50. 510 is a register writing history flag set up by the register writing reserving means 20 and shows whether a writing reservation is made for the register within a block from the past to the present by considering from an instruction branching to the next instruction branching as one block. 520 is a register writing reserved flag set up by the register writing reserving means 20 and shows whether a writing reservation for the register is issued in a block at the present point of time. 530 is an instruction issued flag set up by the instruction issuing means 40 and shows whether an instruction making a writing reservation for the register is issued in a block at the present point of time. As shown in Fig. 3, the register writing history flag 510, register writing reserved flag 520 and instruction issued flag 530 hold flags corresponding to each element of the register and are further provided with two kinds of flag sets for confirmed instructions and unconfirmed instructions in the same form. Terms A side and B side used to differentiate these two kinds of flag registers. Either A side or B side is allotted to one block from one instruction branching to the next instruction branching. This allotment is liberated at the end of all instructions contained in the block and they can be allotted to the next block. 540 is a confirmed instruction group flag showing which side is a confirmed instruction within the two kinds of flag sets in the register writing history flag 510, register writing reserved flag 520 and instruction issued flag 530. Namely, it shows which side of the A side and B side in

execution is an instruction group whose execution has been confirmed. Thereby, if the side where the blocks are allotted is in conformity with the side specified by the confirmed instruction group flag 540, it can be decided that the execution of the block has been confirmed. 550 is an operand reading decision circuit for giving information to the operand access got by the instruction issuing means 40 and prepares reading original information for making the operand reading by either of the branching predictive destination instruction execution result storage means 70 and register 80 and reading propriety information of whether the reading may be carried out from values given by the register writing history flag 510B, register writing reserved flag 520, instruction issued flag 530 and confirmed instruction group flag 540. The decision flow chart of these reading informations are shown in Fig. 4. In Fig. 4, WH represents the register writing history flag 510, WB represents the register writing reserved flag 520, IB represents the instruction issued flag 530, FP represents the confirmed instruction group flag 540, Blk represents the side of the block, rBlk represents the opposite side of the block, and Reg represents the deciding the register number. In particular, the register writing history flag 510, register writing reserved flag 520 and instruction issued flag 530, for example, [flag showing the register number of the flag set on the block side of the register writing reserved flag 520], are expressed in the form of WB(Blk,Reg).

In the instruction processor of the structure as described above, first, actions at the time of processing an instruction whose execution has been confirmed are illustrated to explain the contents of the present invention. The instruction is read from the instruction storage means 10, and a writing into the register for executing the instruction as execution result is reserved by the register writing

reserving means 20. In the reservation operation of reserving the register writing, flags showing the writing register number of the flag set on the block A side are set up for both the register writing history flag 510 and register writing reserved flag 520. In the flags set up here, the register writing history flag 510 is released at the time when the whole instruction till the next branching instruction is ended, and the register writing reserved flag 520 is released at the time when the instruction is ended. The instruction for which the reservation is completed by the register writing reserving means 20 is held in the instruction holding means 30 and waits for the instruction issuance. The instruction held in the instruction holding means 30 gets an operand access in the instruction issuing means 40 and instruction succeeding in it is transferred to the instruction executing means 60. In the operand access executed here, information relating to the reading is received by the operand access controlling means 50. At this time, a decision is made along the flow chart of Fig. 4 in the operand access controlling means 50. First, a comparison of the block and the confirmed instruction group flag 540 (step 101) is made, and it is known that the execution of the block has been confirmed from their conformity. Then, a flag showing the reading register number of the flag set of the instruction issued flag 530 on the block side is decided (step 102). If this flag is in a released state, the operand access to the register 80 is permitted. A flag showing the writing register number of the flag set of the instruction issued flag 530 on the block side is set up simultaneously with issuing an instruction from the instruction issuing means 40. The instruction issued from the instruction issuing means 40 perform an action prescribed by the instruction in the instruction executing means 60 and produces its execution result. The execution result produced in the instruction executing means 60 is stored in the register 80 to

end the execution. At the execution end of the instruction, the flags showing the writing register number of the flag set on the block side are released for both the register writing reserved flag 520 and instruction issued flag 530.

Next, actions in case of executing an instruction which is predicted by a branching prediction mechanism and whose execution has not been confirmed are illustrated. An instruction is read from the instruction storage means 10, and a writing into the register for executing the instruction as execution result is reserved by the register writing reserving means 20. In the reservation operation of the register writing, the flag showing the writing register number of the flag set on the block side are set up for both the register writing history flag 510 and register writing reserved flag 520. In the flags set up here, the register writing history flag 510 is released at a time when the whole instruction till the next branching instruction is ended, and the register writing reserved flag 520 is released at a time when the instruction is ended. The instructions whose reservation are completed by the register writing reserving means 20 got an operand access, and those succeeding in it are transferred to the instruction executing means 60. In the operand access got here, such reading original information that the operand reading is executed by which of the branching predictive destination instruction execution result storage means 70 and the register 80 and such reading propriety information that the reading may be executed are received by the operand access controlling means 50. At this time, a decision is made along the flow chart of Fig. 4 in the operand access controlling means 50. First, make a comparison of the block and the confirmed instruction group flag 540 (step 101), and it is known that the execution of the block has not been confirmed from their inconformity. A flag showing the reading register number of the flag set of the register writing

history flag 510 on the block side is decided (step 103) and is carried to the instruction issuing means 40 so that the operand access is executed from the register 80 if this flag is in a released state and from the branching predictive destination instruction execution result storage means 70 if the flag is in a set state. Moreover, the operand access controlling means 50 originally carries this operand reading original information to the instruction issuing means 40 to prepare operand access permission information. Namely, if it is read from the register 80, a flag showing the reading register number of the flag set of the register writing reserved flag 520 on the opposite side of the block is decided (step 104), if this flag is in a released state, the operand access onto the register is permitted. If it is read from the branching predictive destination instruction execution result storage means 70, a flag showing the register number of the flag set of the instruction issued flag 530 on the block side is decided (step 105), if this flag is in a released state, the operand access onto the branching predictive destination instruction execution result storage means 70 is permitted. A flag showing the writing register number of the flag set of the instruction issued flag 530 on the block side is set up simultaneously with issuing an instruction from the instruction issuing means 40. The instruction issued from the instruction issuing means 40 performs an action prescribed by the instruction in the instruction executing means 60 and produces its execution result. The execution result produced in the instruction executing means 60 is tentatively held in the branching predictive destination instruction execution result storage means 70. At this time, the instruction is regarded to end immediately, and the flags showing the writing register number of the flag set on the block side are released for both the register writing reserved flag 520 and instruction issued flag 530.

Next, actions for confirming a branching destination based on a branching instruction and making it in conformity with a predicted destination are illustrated. If the branching destination of the branching instruction is confirmed, the execution end of instruction before the branching instruction is waited for, and then the execution result tentatively held in the branching predictive destination instruction execution result storage means 70 is transferred to the register 80. After all the transfer is ended, all elements of the branching predictive destination instruction execution result storage means 70 are released. In the register writing history flag 510, all elements of the flag set prescribed by the confirmed instruction group flag 540 are released. Subsequently, the confirmed instruction group flag 540 is inverted.

Then, actions in case that a branching destination based on a branching instruction is confirmed, but it is not in conformity with a predicted branching destination and the branching prediction fails are illustrated. If the branching prediction failure is detected, the input of the former instruction subjected to the branching prediction failure from the instruction storage means 10 is stopped, the end of execution of the former instruction subjected to the branching prediction failure which has been input into the instruction issuing means 40 is waited for, and all elements of the branching predictive destination instruction execution result storage means 70 are released. In all of the register writing history flag 510, register writing reserved flag 520 and instruction issued flag 530, instructions whose branching destination is confirmed from the instruction storage means 10 in which all elements of the flag set on the side opposite to the side prescribed by the confirmed instruction group flag 540 are input and taken as instructions whose execution is confirmed to continue the execution.

Subsequently, an action example is illustrated in a program shown in Fig. 5. The branching instruction in the program is predicted on the side where the branching does not occur. Moreover, the confirmed instruction group flag 540 indicates the A side as initial value. Namely, the A side is used in the flag sets of two kinds A, B in the register writing history flag 510, register writing reserved flag 520 and instruction issued flag 530. First, in the first cycle, an instruction (1) is read from the instruction storage means 10 and turns ON No. 2 flags of the A side of both the register writing history flag 510 and register writing reserved flag 520 corresponding to a writing register r2 of the instruction (1) via the register writing reserving means 20. In the second cycle, an instruction (2) is read from the instruction storage means 10 and turns ON No. 4 flags of the A side of both the register writing history flag 510 and register writing reserved flag 520 corresponding to a writing register r4 of the instruction (2) via the register writing reserving means 20. Simultaneously, the instruction (1) is held by the instruction holding means 30. In the third cycle, an instruction (3) is read from the instruction storage means 10, the register writing reserving means 20 does not make a writing reservation because no register writing for the instruction (3). Simultaneously, the instruction (2) is held by the instruction holding means 30. Simultaneously, the instruction (1) reads the operands r0 and r1 to issue an instruction in the instruction issuing means 40. At this time, the operand access succeeds because No. 0 and No. 1 flags of A side of the instruction issued flag are checked and are OFF. At the time of instruction issuance, No. 2 flags of the A side of both the ? (one flag is missing here, translator) and instruction issued flag 530 corresponding to the writing register r2 of the instruction (1) are turned ON. In the fourth cycle, an instruction (4) is read from the instruction storage

means 10 and turns ON No. 5 flags of the B side of both the register writing history flag 510 and register writing reserved flag 520 corresponding to a writing register r5 of the instruction (4) via the register writing reserving means 20. Simultaneously, the instruction (3) is held by the instruction holding means 30. Simultaneously, the instruction (2) seems to read the operands r2 and r3 in the instruction issuing means 40, but the operand access fails and remains in the instruction issuing means 40 because No. 2 flag of the A side of the instruction issued flag turns ON. A snapshot in the fourth cycle is shown in Fig. 6.

In the fifth cycle, the instruction (1) continues an computing execution, the instruction (2) continues to be held in the instruction issuing means 40, and the instruction (3) continues to be held in the ins-truction holding means 30. The instruction (5) is read from the instruction storage means 10 and turns ON No. 6 flags of the B side of both the register writing history flag 510 and register writing reserved flag 520 corresponding to a writing register r6 of the instruction (5) via the register wri-ting reserving means 20. The instruction (4) decides No. 0 and No. 3 flags of the A side of the register writing history flag 510 corresponding to reading registers r0, r3 of instruction (4) and reads operands from the instruction storage means 10 because the both flags are OFF. Then, No. 0 and No. 3 flags of the A side of the register writing reserved flag 520 are decided and the operand access succeeds because the both are OFF, and the instruction (4) is issued from the instruction issuing means 40. In the sixth cycle, an instruction (6) is input from the instruction storage means 10. Simultaneously, the operands r2 and r3 seem to be read in the instruction issuing means 40, but the operand access fails and the instruction remains in the instruction issuing means 40 because the No. 2 flag of the A side of the instruction issued

flag turns ON. In the seventh cycle, an instruction (7) seems to be input from the instruction storage means 10, but it cannot be input because the instruction (6) remains in register writing reserving means 20. No changes in the eighth cycle. The snap-shot in the eighth cycle is shown in Fig. 7.

In the ninth cycle, a writing of execution result of the instruction (1) into the register 80 occurs, and No. 2 flag of the A side of the register writing reserved flag 520 and the instruction issued flag 530 are turned OFF. Thereby, the operand access of both the instruction (2) and instruction (5) in the instruction issuing means 40 waiting for turning OFF No. 2 flag of the A side of the instruction issued flag 530 succeeds, but only the instruction (2) on the confirmation group side is issued because it becomes a constitution capable of issuing only one instruction simultaneously in this Embodiment. In the tenth cycle, the issuance of the instruction (3) is tried, but it fails in the operand access because No. 4 flag of the A side of the instruction issued flag is turned OFF by the instruction (2) and remains in the instruction issuing means 40. In the tenth cycle, moreover, the issuance of the instruction (5) reserved in the preceding cycle is executed. Thereby, in the eleventh cycle, the instruction (6) proceeds to the instruction issuing means 40 and the instruction (7) proceeds to the register writing reserving means 20. Furthermore, a writing of execution result of the instruction (4) into the branching predictive destination instruction execution result storage means 70 occurs, and No. 5 of the B side of the register writing reserved flag 520 and instruction issued flag 530 are turned OFF. Thereby, as described below, the operand access of the instruction (6) in the instruction issuing means 40 succeeds, and the instruction (6) is issued. Namely, it is known that an instruction is from the branching predictive destination

instruction execution result storage means 70 because No. 5 of the B side of the register writing history flag 510 is turned ON, and the operand access is permitted by turning OFF No. 5 of the B side of the instruction issued flag 530. The snapshot in the eleventh cycle is shown in Fig. 8.

No changes occur between the twelfth cycle to the fourteenth cycle. In the fifteenth cycle, a writing of execution result of the instruction (2) into the register 80 occurs, and No. 4 of the A side of the register writing reserved flag 520 and the instruction issued flag 530 are turned OFF. Thereby, the instruction (3) in the instruction issuing means 40 waiting for turning OFF No. 4 of the A side of the instruction issued flag 530 is issued. A writing of execution result of the instruction (2) into the branching predictive destination instruction execution result storage means 70 occurs in the sixteenth cycle and then a writing of execution result of the instruction (6) into the branching predictive destination instruction execution result storage means 70 occurs in the seventeenth cycle, and No. 6, No. 7 of the B side of the register writing reserved flag 520 and the instruction issued flag 530 are turned OFF. Thereby, the instruction (7) in the instruction issuing means 40 waiting for turning OFF No. 7 of the B side of the instruction issued flag 530 is issued in the seventeenth cycle. No changes occur in the eighteenth cycle. In the nineteenth cycle, a transfer from the branching predictive destination instruction execution result storage means 70 to the register 80 is executed because the execution of instruction (3) is ended and confirmed by the predicted branching destination. At this time, execution results r5, r6, r7 based on the instructions (4), (5), (6) exist in the branching predictive destination instruction execution result storage means 70, and the transfer of these three is executed. After the transfer, the branching predictive destination

instruction execution result storage means 70 is released. Then, all elements of the A side of the register writing history flag 510 are released, and values prescribed by the confirmed instruction group flag 540 are inverted from the A side to the B side. Thereby, an instruction (8) is input from the instruction storage means 10 because the A side can be newly used in the twentieth cycle. The snapshot in the twentieth cycle is shown in Fig. 9.

In the twenty-first cycle, the execution of instruction (7) is ended and confirmed by the predicted branching destination. However, the transfer from the branching predictive destination instruction execution result storage means 70 to the register 80 is not executed at this time because a writing does not occur in the branching predictive destination instruction execution result storage means 70. Then, all elements of the B side of the register writing history flag 510 are released, and values prescribed by the confirmed instruction group flag 540 are inverted from the B side to the A side. The instruction (8) is issued in the twenty-first cycle and ended in the twenty-sixth cycle to end the program. A time chart from the start to the end of program is shown in Fig. 10.

The processing course of the instructions in the instruction processor based on the invention of this application were illustrated above.

Next, another Embodiment in case of using two instruction holding means 31, 32 in place of one instruction holding means 30 of Fig. 1 is illustrated by Fig. 11.

Fig. 11 is one Embodiment of the instruction processor based on the present invention. It becomes the same constitution as the Embodiment of the instruction processor based on the present invention of aforesaid Fig. 1 except that the two instruction holding means 31, 32 are provided in the diagram. In this constitution, instructions

in which the reservation is completed by the register writing reserving means 20 are held in an instruction holding means A or B with branching instructions as marks. For example, in an instruction group held in the instruction holding means A at the beginning, an instruction next to a branching instruction or thereafter is held in the instruction holding means B, and an instruction next to the next branching instruction or thereafter is held in the instruction holding means A. Thereby, the instruction (7) could not be input from the instruction storage means 10 until the tenth cycle in the example of program of Fig. 5,, but the input is enabled in the sixth cycle in the present invention wherein two instruction holding means are used. Still another Embodiment of the present invention in the case of using two registers is illustrated by Fig. 12.

The instruction processor of the present invention of Fig. 2 is provided with two registers 81, 82 and a transfer means therebetween, and it becomes the same constitution as the Embodiment of the instruction processor based on the present invention of aforesaid Fig. 1 except that the branching predictive destination instruction execution result storage means 70 is omitted. In this constitution, if a result obtained in the instruction executing means 60 is a result based on instructions belonging to a confirmed group, it is written into the both registers 81, 82, but if the result is a result based on instructions belonging to an unconfirmed group, it is held in either register A or B with branching instructions as marks. For example, in case a result based on some unconfirmed instruction group is held in the register A, the result based on this instruction group is written in both the registers A, B from the instant when the preceding branching prediction succeeds to confirm this instruction group. Then, a result of an instruction next to the branching instruction and thereafter is held only in the register B until the execution is confirmed, and

a result based on the instruction group is written in both the registers A, B from the instant when the group is confirmed. A result of instruction next to the next branching instruction and thereafter is held in the register A until the execution is confirmed, and this is repeated hereinafter. A flag register of the form of aforesaid Fig. 3, called a register written flag, is provided in a between-register transfer means 90, when the results are written only in one register in the above operation procedure, the register written flag of the register number is turned ON on an equivalent side (i. e., the A side if it is register A and the B side if it is register B). Then, at the time that a branching instruction for determining the execution propriety of an instruction is branched on the prediction success side, a register of the register number on the block side where the register written flag is turned ON is copied in the same register number on the opposite side. The register written flag is turned OFF after this copying. Conversely, when it is branched on the prediction failure side, a register of the register number on the side opposite to the block side where the register written flag is turned ON is copied in the same register number on the block side where the register written flag is turned ON. The register written flag is turned OFF after this copying. Thereby, the execution result of the branching predictive destination is written in one-side register before the branching destination is confirmed, and original values can be restored by copying the result on the opposite A side at the time of branching success and copying the values on the opposite A side at the time of branching prediction failure.

A flow chart for determining the propriety of register reading and the judgement of which register it should be read from is shown in Fig. 13. Symbols used are same as Fig. 4.

Next, actions in the Embodiment of Fig. 12 are illustrated in the program shown in afore-said Fig. 5. As compared with the case of aforesaid Fig. 1, only actions of the operand reading and the execution result writing are different, therefore only this section is illustrated. First, the r2 of A side can be read according to the flow chart of Fig. 13 because the instruction (1) is on the confirmation group side (Fig. 13, step 201) and the instruction for writing of r2 is not in execution (step 204). Moreover, the instruction (1) is written in the both side registers because it is a confirmation group in writing. The next instruction (2) makes the operand reading from the register of the block side, i. e., the A side according to the flow chart of Fig. 13 because it is still the confirmation group side. At this time, the preceding instruction (1) has already reserved the writing of r2 (i. e., IB(Blk,Reg ON)), therefore the operand reading is reserved until the ninth cycle where the instruction (1) is ended. The instruction (1) seems to read the r2 like the instruction (2), but it is an unconfirmed group side (Fig. 13, step 201) and the writing into r2 based on the block is not made (step 202), therefore an operand reading is tried from the register on the opposite side of the block, i. e., the A side. However, the operand reading is reserved until the instruction (1) is ended because the instruction (1) has already reserved the writing of r2 (the issuance of instruction (5) is in the tenth cycle because the instruction (2) is issued in the ninth cycle). Moreover, the writing of result is made only into the register of the block side, i. e., the B side. Other instructions also take the same actions.

Next, one Embodiment in case of using two instruction issuing means is illustrated by Fig. 14, Fig. 15. Fig. 14 is one Embodiment of the instruction processor based on the present invention. In the diagram, it becomes the same constitution as Embodiment 1 of the instruction

processor based on the present invention of aforesaid Fig. 1 except that two instruction issuing means 41, 42 are provided, simultaneous issuance of plural instructions are enabled, and an operand access controlling means 150, an instruction executing means 160, a branching predictive destination instruction execution result storage means 170 and a register 180 are simultaneously accessible. Fig. 15 shows principal parts of the operand access controlling means 150. In the diagram, the flags become simultaneously accessible. In the diagram, it becomes the same constitution as aforesaid Fig. 2 except that two operand reading decision circuits are provided.

Next, actions in the Embodiment of Fig. 14 are illustrated in a program showing in afore-said Fig. 5. As compared with the case of aforesaid Fig. 1, only action timings of the instruction issuing means are different, therefore only this section is illustrated. A writing of execution result into the register 80 occurs in the ninth cycle, No. 2 flags of the A side of the register writing reserved flag 520 and instruction issued flag 530 are turned OFF. Thereby, the operand access of both instruction (2) and instruction (5) in the instruction issuing means waiting for turning OFF No. 2 of the A side of instruction issued flag 530 succeeds. It becomes a constitution where only one instruction can be issued in the Embodiment of Fig. 1, but both the instruction (2) and instruction (5) are issued in the ninth cycle because two instruction issuing means can be provided and the two instructions can be simultaneously issued in the Embodiment of Fig. 14. Actions of other instructions are same as the case of Embodiment of Fig. 1.

Moreover, the instruction processor may also be constructed by properly combining the embodiments of Fig. 11, Fig. 12 and Fig. 14.

Namely, the instruction processor may also be constructed so as to contain at least two means in the plural instruction holding means A, B, registers A, B and instruction issuing means A, B.

[Efficacy of the invention]

As described above, this invention enables using execution results based on an instruction of a pre-dicted branching destination before confirmation as an operand, suppressing the delay of an instruction execution start caused by confirmation waiting of a branching instruction and performing high speed processing while executing an instruction group before the branching instruction and an instruction group of a predicted branching destination simultaneously and as far as possible.

Brief description of the drawings

Fig. 1 is a system diagram of one Embodiment of the present invention,

Fig. 2 is a diagram showing one Embodiment of an operand access controlling means of Fig. 1.

Fig. 3 is a diagram showing the constitution of a register writing history flag of Fig. 2, a register writing reserved flag, instruction issued flag and a register written flag of between-register transfer means of Fig. 12.

Fig. 4 is a flow chart for reading and permission decision in the operand access controlling means of Fig. 1.

Fig. 5 is a program used for showing an action example.

Figs. 6, 7, 8, 9 are snapshots of the fourth, eighth, eleventh and twentieth cycles of the program action example based on Fig. 5, respectively.

Fig. 10 is a time chart of the program action example based on Fig. 5.

Fig. 11 is a system diagram showing one Embodiment of the second invention of this application.

Fig. 12 is a system diagram showing one Embodiment of the third invention of this application.

Fig. 13 is a flow chart for reading permission decision in an operand access controlling means of Fig. 12.

Fig. 14 is a system diagram showing one Embodiment of the fourth invention of this application.

Fig. 15 is a diagram showing one Embodiment of an operand access controlling means of Fig. 14.

Fig. 16 is a system diagram based on the prior art.

In the drawings,

10 instruction storage means
20 register writing reserving means
31, 32, 33 instruction holding means
40, 41, 42 instruction issuing means
50 operand access controlling means
60 instruction executing means
70 branching predictive destination instruction execution
result storage means
80, 81, 82 registers
90 between-register transfer means
150 operand access controlling means
160 instruction executing means
170 branching predictive destination instruction execution
result storage means
180 register
510 register writing history flag
520 register writing reserved flag
530 instruction issued flag
540 confirmed instruction group flag
550 operand reading decision circuit
561 signal line transmitting branching information of branching
instructions
562 signal line transmitting end of instructions
563 signal line transmitting writing reservation information
of instructions into register
564 signal line transmitting issuance information of
instructions register
565 signal line transmitting propriety of operand reading

1510 register writing history flag
1520 register writing reserved flag
1530 instruction issued flag
1540 confirmed instruction group flag

Fig. 1

10 instruction storage means
20 register writing reserving means
30 instruction holding means
40 instruction issuing means
50 operand access controlling means
60 instruction executing means
70 branching predictive destination instruction execution
result storage means
80 register

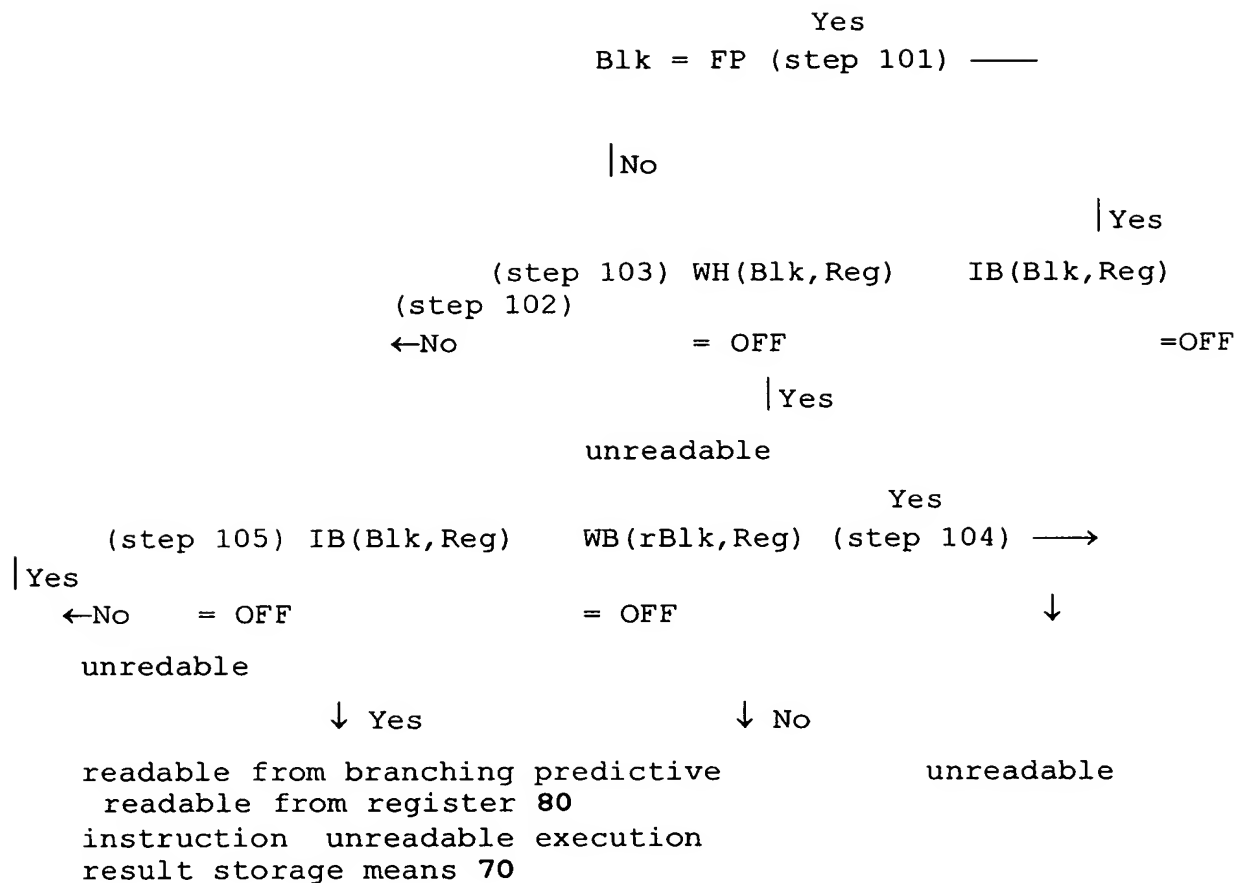
Fig. 2

510 register writing history flag
520 register writing reserved flag
530 instruction issued flag
540 confirmed instruction group flag
550 operand reading decision circuit

Fig. 3

N: number of elements of register

Fig. 4



WH : register writing history flag 510
 WB : register writing reserved flag 520
 IB : instruction issued flag 530
 FP : confirmed instruction group flag
 Blk : side of the block

rBlk : opposite side of the block
 Reg : the register number

Fig. 5

	Execution time after instruction issuance
Instruction (1): multiplication instruction 6 cycle	Mlt r0*r1 = r2
Instruction (2): multiplication instruction 6 cycle	Mlt r2*r3 → r4
Instruction (3): branching instruction 0) 4 cycle	Brch To x 1 on (r4 =
Instruction (4): multiplication instruction 6 cycle	Mlt r0*r3 → r5
Instruction (5): multiplication instruction 6 cycle	Mlt r2*r3 → r6
Instruction (6): multiplication instruction 6 cycle	Mlt r0*r6 → r7
Instruction (7): multiplication instruction 0) 4 cycle	Brch To x 2 on (r7 =
Instruction (8): addition instruction 5 cycle	Add r0 + r3 → r0

```

10      instruction storage means
20      register writing reserving means      instruction (4)
30      instruction holding means      instruction (3)
40      instruction issuing means instruction (2)
50      operand access controlling means
60      instruction executing means      instruction (1)
70      branching predictive destination instruction execution
result storage means
80      register

```

```

10      instruction storage means
20      register writing reserving means      instruction (6)
30      instruction holding means      instruction (3)
40      instruction issuing means      instruction (5)
                                     instruction (2)
50      operand access controlling means
60      instruction executing means      instruction (4)
                                     instruction (1)
70      branching predictive destination instruction execution
result storage means
80      register

```

Fig. 8

10 instruction storage means
20 register writing reserving means
30 instruction holding means
40 instruction issuing means instruction (7)
 instruction (3)
50 operand access controlling means
60 instruction executing means instruction (6)
 instruction (5)
 instruction (2)
70 branching predictive destination instruction execution
result storage means
 instruction (4)
80 register instruction (1)

Fig. 9

10 instruction storage means
20 register writing reserving means instruction (6)

30 instruction holding means
40 instruction issuing means
50 operand access controlling means
60 instruction executing means instruction (7)
70 branching predictive destination instruction execution
result storage means
80 register instruction (1)

 instruction (2)
 instruction (4)
 instruction (5)
 instruction (6)

Fig. 10

Instruction (1)
Instruction (2)
Instruction (3)
Instruction (4)
Instruction (5)
Instruction (6)
Instruction (7)
Instruction (8)

issuing means **40**

Instruction storage **10** - Instruction

Instruction executing means **60**

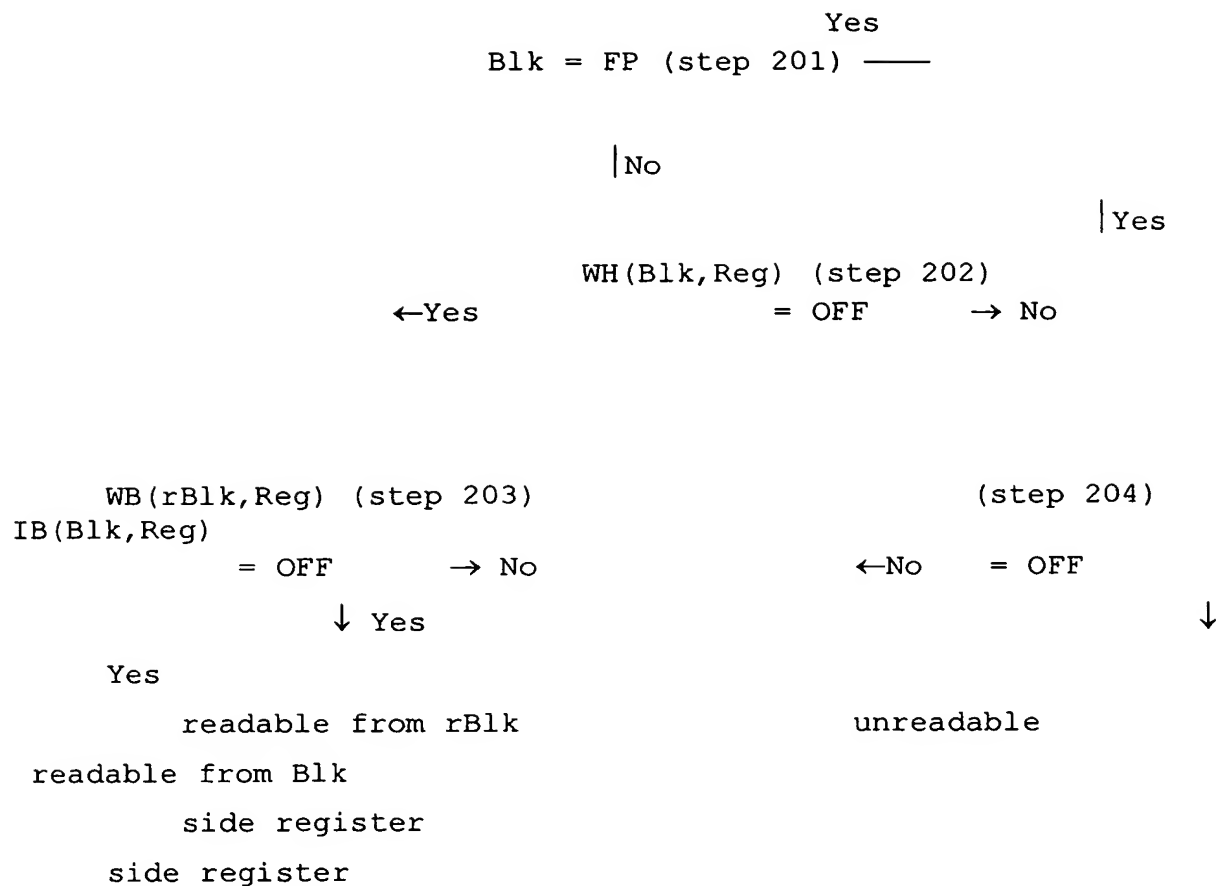
Fig. 11

10	instruction storage means
20	register writing reserving means
31	instruction holding means A
32	instruction holding means B
40	instruction issuing means
50	operand access controlling means
60	instruction executing means
70	branching predictive destination instruction execution
	result storage means
80	register

Fig. 12

10	instruction storage means
20	register writing reserving means
30	instruction holding means
40	instruction issuing means
50	operand access controlling means
60	instruction executing means
81	register A
82	register B
90	between-register transfer means

Fig. 13



WH : register writing history flag 510
 WB : register writing reserved flag 520
 IB : instruction issued flag 530
 FP : confirmed instruction group flag
 Blk : side of the block
 rBlk : opposite side of the block
 Reg : the register number

Fig. 14

10	instruction storage means
20	register writing reserving means
30	instruction holding means
41	instruction issuing means A
42	instruction issuing means B
150	operand access controlling means
160	instruction executing means
170	branching predictive destination instruction execution
	result storage means
180	register

Fig. 15

551	operand reading decision circuit A
552	operand reading decision circuit B
1510	register writing history flag
1520	register writing reserved flag
1530	instruction issued flag
1540	confirmed instruction group flag

Fig. 16

10 instruction storage means
20 register writing reserving means
40' instruction issuing means
50 operand access controlling means
60 instruction executing means
70' branching predictive destination instruction execution
result storage means
80 register

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.